Zero-shot Generalist Graph Anomaly Detection with Unified Neighborhood Prompts

Chaoxi Niu[1†] , **Hezhe Qiao**[2†] , **Changlu Chen**[3] , **Ling Chen**[1] and **Guansong Pang**[2*]

[1] AAII, University of Technology Sydney, Sydney, Australia
[2] School of Computing and Information Systems, Singapore Management University, Singapore
[3] Faculty of Data Science, City University of Macau, Macau, China
chaoxi.niu@student.uts.edu.au, hezheqiao.2022@phdcs.smu.edu.sg, clchen@cityu.edu.mo,
ling.chen@uts.edu.au, gspang@smu.edu.sg

Abstract

Graph anomaly detection (GAD), which aims to identify nodes in a graph that significantly deviate from normal patterns, plays a crucial role in broad application domains. However, existing GAD methods are one-model-for-one-dataset approaches, *i.e.*, training a separate model for each graph dataset. This largely limits their applicability in real-world scenarios. To overcome this limitation, we propose a novel zero-shot generalist GAD approach **UNPrompt** that trains a one-for-all detection model, requiring the training of one GAD model on a single graph dataset and then effectively generalizing to detect anomalies in other graph datasets without any retraining or fine-tuning. The key insight in UNPrompt is that i) the predictability of latent node attributes can serve as a generalized anomaly measure and ii) generalized normal and abnormal graph patterns can be learned via latent node attribute prediction in a properly normalized node attribute space. UNPrompt achieves a generalist mode for GAD through two main modules: one module aligns the dimensionality and semantics of node attributes across different graphs via coordinate-wise normalization, while another module learns generalized neighborhood prompts that support the use of latent node attribute predictability as an anomaly score across different datasets. Extensive experiments on real-world GAD datasets show that UN-Prompt significantly outperforms diverse competing methods under the generalist GAD setting, and it also has strong superiority under the one-model-for-one-dataset setting. Code is available at https://github.com/mala-lab/UNPrompt.

1 Introduction

Graph anomaly detection (GAD) has attracted extensive research attention in recent years [Pang *et al.*, 2021; Qiao *et al.*, 2024] due to the broad applications in various domains such as spam review detection in online shopping networks

---

*Corresponding author: Guansong Pang (gspang@smu.edu.sg).

[McAuley and Leskovec, 2013; Rayana and Akoglu, 2015] and malicious user detection in social networks [Yang *et al.*, 2019]. To handle high-dimensional node attributes and complex structural relations between nodes, graph neural networks (GNNs) [Kipf and Welling, 2016; Wu *et al.*, 2020] have been widely exploited for GAD due to their strong ability to integrate the node attributes and graph structures. These methods can be roughly divided into two categories, *i.e.*, supervised and unsupervised methods. The former formulates GAD as a binary classification problem and aims to capture anomaly patterns under the guidance of labels [Tang *et al.*, 2022; Peng *et al.*, 2018; Gao *et al.*, 2023]. By contrast, due to the difficulty of obtaining these class labels, the latter category takes the unsupervised approach that aims to learn normal graph patterns, *e.g.*, via data reconstruction or other proxy learning tasks that are related to GAD [Qiao and Pang, 2023; Liu *et al.*, 2021b; Ding *et al.*, 2019; Huang *et al.*, 2022].

Despite their remarkable detection performance, these methods need to train a dataset-specific GAD model for each graph dataset. This one-model-for-one-dataset paradigm limits their applicability in real-world scenarios since training a model from scratch can incur extensive computation costs and require a large amount of labeled data for supervised GAD methods on each dataset [Liu *et al.*, 2024; Qiao *et al.*, 2024]. Training on a target graph may not even be possible due to data privacy protection and regulation. To address this limitation, a new one-for-all anomaly detection (AD) paradigm, called generalist anomaly detection [Zhu and Pang, 2024; Zhou *et al.*, 2024], has been proposed for image AD with the emergence of foundation models such as CLIP [Radford *et al.*, 2021]. This new direction aims to learn a generalist detection model on auxiliary datasets so that it can generalize to detect anomalies effectively in diverse target datasets without any re-training or fine-tuning. This paper explores this direction in the area of GAD.

Compared to image AD, there are some unique challenges for learning generalist models for GAD. First, unlike image data where raw features are in the same RGB space, the node attributes of graphs from different applications and domains can differ significantly in node attribute dimensionality and semantics. For example, as a shopping network dataset, Amazon contains the relationships between users and reviews, and the node attribute dimensionality is 25. Differently, Face-
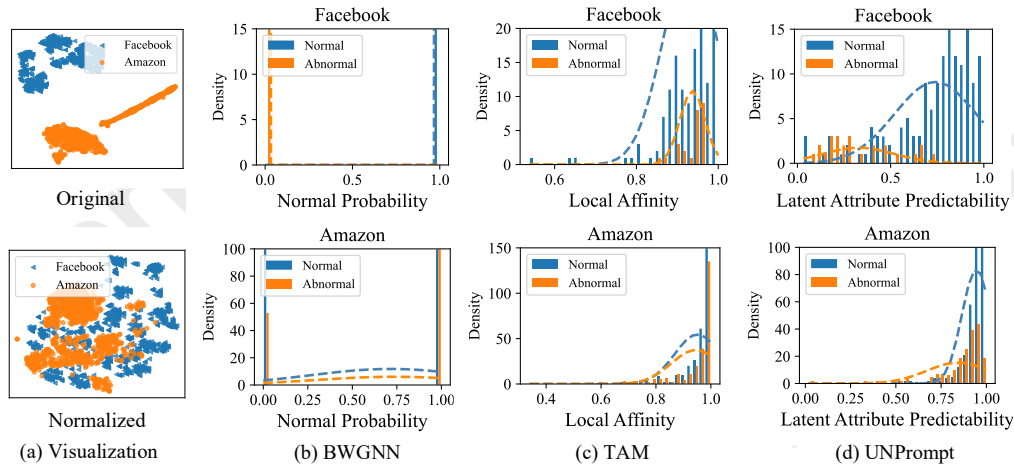
Figure 1: **(a)** Visualization of two popular GAD datasets: Facebook and Amazon, where the node attributes are unified into a common semantic space via our proposed normalization compared to the original heterogeneous raw attributes. **(b)-(d)** The anomaly scores of BWGNN (normal probability), TAM (local affinity), and UNPrompt (latent attribute predictability) on the two datasets, where the methods are all trained on Facebook and tested on Amazon under the zero-shot setting. It is clear that BWGNN and TAM struggle to generalize from Facebook to Amazon, while UNPrompt can learn well to generalize across the datasets.

book, a social network dataset, describes relationships between users with 576-dimensional attributes. Second, generalist AD models on image data rely on the superior generalizability learned in large visual-language models (VLMs) through pre-training on web-scale image-text-aligned data [Zhu and Pang, 2024; Zhou *et al.*, 2024], whereas there are no such foundation models for graphs [Liu *et al.*, 2023a]. Therefore, the key question here is: *can we learn generalist models for GAD on graph data with heterogeneous node attributes and structure without the support of foundation models?*

To address these challenges, we propose **UNPrompt**, a novel generalist GAD approach that learns *Unified Neighborhood Prompts* on a single auxiliary graph dataset and then effectively generalizes to directly detect anomalies in other graph datasets under a **zero-shot** setting. The key insight in UNPrompt is that i) the predictability of latent node attributes can serve as a generalized anomaly measure and ii) graph-agnostic normal and abnormal patterns can be learned via latent node attribute prediction in a properly normalized node attribute space. UNPrompt achieves this through two main modules, including *coordinate-wise normalization-based node attribute unification* and *neighborhood prompt learning*. The former module aligns the dimensionality of node attributes across graphs and transforms the semantics into a common space via coordinate-wise normalization, as shown in Figure 1(a). In this way, the diverse distributions of node attributes are calibrated into the same semantic space. On the other hand, the latter module focuses on modeling graph-agnostic normal and abnormal patterns across different graph datasets. This is achieved in UNPrompt by learning generalized graph prompts in the normalized attributes of the neighbors of a target node via a latent node attribute prediction task. In doing so, UNPrompt, being trained on a single graph with a small GNN, can yield effective anomaly scores for detecting anomalous nodes in diverse unseen graphs without any re-training, as shown in Figure 1(b)-(d).

Overall, the main contributions of this paper are summarized as follows. **(1)** We propose a novel zero-shot generalist GAD approach, UNPrompt. To the best of our knowledge, this is the first method that exhibits effective zero-shot GAD performance across various graph datasets. There is concurrent work on generalist GAD [Liu *et al.*, 2024], but it can only work under a few-shot setting. **(2)** By unifying the heterogeneous distributions in the node attributes across different graphs, we further introduce a novel neighborhood prompt learning module that utilizes a neighborhood-based latent node attribute prediction task to learn generalized prompts. This enables the zero-shot GAD of UNPrompt across different graphs. **(3)** Extensive experiments on real-world GAD datasets show that UNPrompt significantly outperforms state-of-the-art competing methods under the zero-shot generalist GAD. **(4)** We show that UNPrompt can also work in the conventional one-model-for-one-dataset setting, outperforming state-of-the-art models in this popular GAD setting.

2 Related Work

Graph Anomaly Detection. Existing GAD methods can be roughly categorized into unsupervised and supervised approaches [Qiao *et al.*, 2024]. The unsupervised methods are typically built using data reconstruction, self-supervised learning, and learnable graph anomaly measures [Qiao *et al.*, 2024; Liu *et al.*, 2022]. The reconstruction-based approaches like DOMINANT [Ding *et al.*, 2019] and AnomalyDAE [Fan *et al.*, 2020] aim to capture the normal patterns in the graph, where the reconstruction error in both graph structure and attributes is utilized as the anomaly score. CoLA [Liu *et al.*, 2021b] and SL-GAD [Zheng *et al.*, 2021] are representative self-supervised learning methods assuming that normality is reflected in the relationship between the target node and its contextual nodes. The graph anomaly measure methods typically leverage the graph structure-aware anomaly measures to

learn intrinsic normal patterns for GAD, such as node affinity in TAM [Qiao and Pang, 2023]. In contrast to the unsupervised approaches, the supervised approaches have shown substantially better detection performance in recent years due to the incorporation of labeled anomaly data [Liu *et al.*, 2021a; Chai *et al.*, 2022]. Most supervised methods concentrate on the design of propagation mechanisms and spectral feature transformations to address the notorious over-smoothing issues [Tang *et al.*, 2022; Gao *et al.*, 2023; Chai *et al.*, 2022]. Although both approaches can be adapted for zero-shot GAD by directly applying the trained GAD models to the target datasets, they struggle to capture generalized normal and abnormal patterns for GAD across different graph datasets. There are some works on cross-domain GAD [Ding *et al.*, 2021; Wang *et al.*, 2023] that aim to transfer knowledge from a labeled graph dataset to a target dataset, but it is a fundamentally different problem from generalist GAD since cross-domain GAD requires training on both source and target graph datasets.

Graph Prompt Learning. Prompt learning, initially developed in natural language processing, seeks to adapt large-scale pre-trained models to different downstream tasks by incorporating learnable prompts while keeping the pre-trained models frozen [Liu *et al.*, 2023b]. Specifically, it designs task-specific prompts capturing the knowledge of the corresponding tasks and enhances the compatibility between inputs and pre-trained models to improve the pre-trained models in downstream tasks. Recently, prompt learning has been explored in graphs to unify multiple graph tasks [Sun *et al.*, 2023; Liu *et al.*, 2023c] or improve the transferability of graph models on the datasets across the different domains [Li *et al.*, 2024; Zhao *et al.*, 2024] which optimize the prompts with labeled data of various downstream tasks [Fang *et al.*, 2024; Liu *et al.*, 2023c]. Although being effective in popular graph learning tasks like node classification and link prediction, they are inapplicable to generalist GAD due to the unsupervised nature and/or irregular distributions of anomalies.

Generalist Anomaly Detection. Generalist AD has been very recently emerging as a promising solution to tackle sample efficiency and model generalization problems in AD. There have been a few studies on non-graph data that have large pre-trained models to support the generalized pattern learning, such as image generalist AD [Zhou *et al.*, 2024; Zhu and Pang, 2024]. However, it is a very challenging task for graph data due to the lack of such pre-trained models. Recently, a concurrent approach, ARC [Liu *et al.*, 2024], introduces an effective framework that leverages in-context learning to achieve generalist GAD without relying on large pre-trained GNNs. Unlike ARC which focuses on a few-shot GAD setting, *i.e.*, requiring the availability of some labeled nodes in the target testing graph dataset, we tackle a zero-shot GAD setting, assuming no access to any labeled data during inference stages.

3 Methodology

3.1 Preliminaries

Notations. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an attributed graph with N nodes, where $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ represents the node set

and $\mathcal{E}$ is the edge set. The attributes of nodes can be denoted as $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times d}$ and the edges between nodes can be presented by an adjacency matrix $A \in \{0, 1\}^{N \times N}$ with $A_{ij} = 1$ if there is an edge between v_i and v_j and $A_{ij} = 0$ otherwise. For simplicity, the graph can be represented as $\mathcal{G} = (A, X)$. In GAD, the node set can be divided into a set of normal nodes $\mathcal{V}_n$ and a set of anomalous nodes $\mathcal{V}_a$. Typically, the number of normal nodes is significantly larger than the anomalous nodes, *i.e.*, $|\mathcal{V}_n| \gg |\mathcal{V}_a|$. Moreover, the anomaly labels can be denoted as $\mathbf{y} \in \{0, 1\}^N$ with $\mathbf{y}_i = 1$ if $v_i \in \mathcal{V}_a$ and $\mathbf{y}_i = 0$ otherwise.

Conventional GAD. Conventional GAD typically focuses on model training and anomaly detection on the same graph. Specifically, given a graph $\mathcal{G}$, an anomaly scoring model $f : \mathcal{G} \to \mathbb{R}$ is optimized on $\mathcal{G}$ in a supervised or unsupervised manner. Then, the model is used to detect anomalies within the same graph. The model is expected to generate higher anomaly scores for abnormal nodes than normal nodes, *i.e.*, $f(v_i) < f(v_j)$ if $v_i \in \mathcal{V}_n$ and $v_j \in \mathcal{V}_a$.

Generalist GAD. Generalist GAD aims to learn a generalist model f on a single training graph so that f can be directly applied to different target graphs across diverse domains without any fine-tuning or re-training. More specifically, the model is optimized on $\mathcal{G}_{\text{train}}$ with the corresponding anomaly labels $\mathbf{y}_{\text{train}}$. After model optimization, the learned f is utilized to detect anomalies within different unseen target graphs $\mathcal{T}_{\text{test}} = \{\mathcal{G}_{\text{test}}^{(1)}, \ldots, \mathcal{G}_{\text{test}}^{(n)}\}$ which has heterogeneous attributes and/or structures to $\mathcal{G}_{\text{train}}$, *i.e.*, $\mathcal{G}_{\text{train}} \cap \mathcal{T}_{\text{test}} = \emptyset$. Depending on whether labeled nodes of the target graph are provided during inference, the generalist GAD problem can be further divided into two categories, *i.e.*, **few-shot** and **zero-shot** settings. We focus on the zero-shot setting where the generalist models cannot get access to any labeled data of the testing graphs during both training and inference.

3.2 Overview of UNPrompt

The framework of UNPrompt is illustrated in Figure 2, which consists of two main modules, coordinate-wise normalization-based node attribute unification and neighborhood prompt learning. For all graphs, the attribute unification aligns the dimensionality of node attributes and transforms the semantics into a common space. Then, in the normalized space, the generalized latent attribute prediction task is performed with the neighborhood prompts to learn generalized GAD patterns. Specifically, UNPrompt aims to maximize the predictability of the latent attributes of normal nodes while minimizing those of abnormal nodes. In this paper, we evaluate the predictability via the similarity. In doing so, the graph-agnostic normal and abnormal patterns are incorporated into the prompts. During inference, the target graph is directly fed into the learned models after node attribute unification without any re-training or labeled nodes of the graph. For each node, the predictability of latent node attributes is directly used as the normal score for final anomaly detection.

3.3 Node Attribute Unification

Graphs from different distributions and domains significantly differ in the dimensionality and semantics of node attributes.
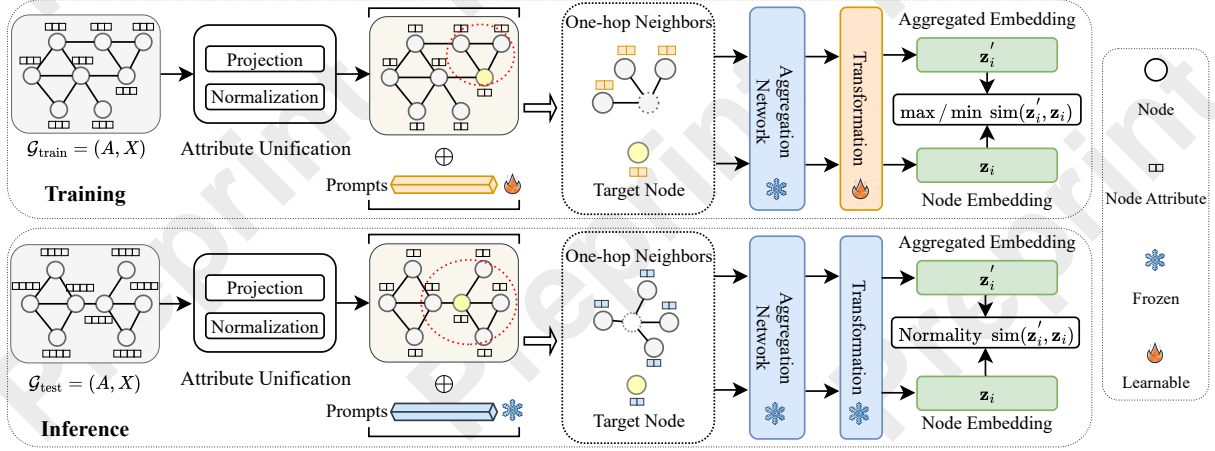
Figure 2: Overview of UNPrompt. During training, the neighborhood prompts are optimized to capture generalized patterns by maximizing the predictability of the latent attributes of normal nodes while minimizing that of abnormal nodes. At the inference stage, the learned prompts are directly applied to the testing nodes, and the latent attribute predictability is used for GAD.

The premise of developing a generalist GAD model is to unify the dimensionality and semantics of node attributes into the same space. To address this issue, we propose a simple yet effective node attribute unification module that consists of feature projection and coordinate-wise normalization.

Feature Projection. To address the inconsistent attribute dimensions across graphs, various feature projection methods can be utilized, such as singular value decomposition [Stewart, 1993] (SVD) and principal component analysis [Abdi and Williams, 2010] (PCA). Formally, given the attribute matrix $X^{(i)} \in \mathbb{R}^{N^{(i)} \times d^{(i)}}$ of any graph $\mathcal{G}^{(i)}$ from $\mathcal{G}_{\text{train}} \cup \mathcal{T}_{\text{test}}$, we transform it into $\tilde{X}^{(i)} \in \mathbb{R}^{N^{(i)} \times d'}$ with the common dimensionality of d',

$$X^{(i)} \in \mathbb{R}^{N^{(i)} \times d^{(i)}} \xrightarrow[\text{Projection}]{\text{Feature}} \tilde{X}^{(i)} \in \mathbb{R}^{N^{(i)} \times d'} . \quad (1)$$

Coordinate-wise Normalization. Besides the inconsistent attribute dimensionality, the divergent node attribute semantics and distributions of different graphs pose significant challenges for generalist GAD. A recent study [Li *et al.*, 2023] has demonstrated that semantic differences across datasets are mainly reflected in the distribution shifts and calibrating the distributions into a common frame helps learn more generalized AD models. Inspired by this, we propose to use coordinate-wise normalization to align the semantics and unify the distributions across graphs. Specifically, the transformed attribute matrix $\tilde{X}^{(i)}$ is shifted and re-scaled to have mean zeros and variance ones via the following equation:

$$\bar{X}^{(i)} = \frac{\tilde{X}^{(i)} - \boldsymbol{\mu}^{(i)}}{\boldsymbol{\sigma}^{(i)}} , \quad (2)$$

where $\boldsymbol{\mu}^{(i)} = [\mu_1^{(i)}, \ldots, \mu_{d'}^{(i)}]$ and $\boldsymbol{\sigma}^{(i)} = [\sigma_1^{(i)}, \ldots, \sigma_{d'}^{(i)}]$ are the coordinate-wise mean and variance of $\tilde{X}^{(i)}$ of the graph $\mathcal{G}^{(i)}$. In this way, the distributions of normalized attributes along each dimension are the same within and across graphs, as shown in Figure 1(a). This helps to capture the generalized normal and abnormal patterns for generalist GAD.

3.4 Neighborhood Prompt Learning

Latent Node Attribute Predictability as Anomaly Score. In this paper, we reveal that the predictability of latent node attributes can serve as a generalized anomaly measure, and highly generalized normal and abnormal graph patterns can be learned via latent node attribute prediction in the normalized attribute space with neighborhood prompts. The key intuition of this anomaly measure is that normal nodes tend to have more connections with normal nodes of similar attributes due to prevalent graph homophily relations, resulting in a more homogeneous neighborhood in the normal nodes [Qiao and Pang, 2023]. By contrast, the presence of anomalous connections and/or attributes makes abnormal nodes deviate significantly from their neighbors. Therefore, for a target node, its latent attributes (*i.e.*, node embedding) are more predictable based on the latent attributes of its neighbors if the node is a normal node, compared to abnormal nodes. The neighborhood-based latent attribute prediction is thus used to measure the normality for GAD. As shown in our experiments (see Figures 1(b)-(d) and Tables 1), it is a generalized anomaly measure that works effectively across graphs. However, due to the existence of irrelevant and noisy attribute information in the original attribute space, the attribute prediction is not as effective as expected in the simply projected space. To address this issue, we propose to learn discriminative prompts via the latent attribute prediction task to enhance the effectiveness of this anomaly measure.

To achieve this, we first design a simple GNN g, a neighborhood aggregation network, to generate the aggregated neighborhood embedding of each target node. Specifically, given a graph $\mathcal{G} = (A, \bar{X})$, the aggregated neighborhood embeddings for each node are obtained as follows:

$$\tilde{Z} = g(\mathcal{G}) = \tilde{A}\bar{X}W , \quad (3)$$

where $\tilde{Z}$ is the aggregated representation of neighbors, $\tilde{A} = (D)^{-1}A$ is the normalized adjacency matrix with D being a diagonal matrix and its elements $D_{kk} = \sum_j A_{kj}$, and W is the learnable parameters. Compared to conventional GNNs

such as GCN [Kipf and Welling, 2016] and SGC [Wu *et al.*, 2019], we do not require $\tilde{A}$ to be self-looped and symmetrically normalized as we aim to obtain the aggregated representation of all the neighbors for each node. To design the latent node attribute prediction task, we further obtain the latent attributes of each node as follows:

$$Z = \bar{X}W \,, \tag{4}$$

where Z serves as the prediction ground truth for the latent attribute prediction task. The adjacency matrix A is discarded to avoid carrying neighborhood-based attribute information into Z which would lead to ground truth leakage in this prediction task. We further propose to utilize the cosine similarity to measure this neighborhood-based latent attribute predictability for each node:

$$s_i = \text{sim}(\mathbf{z}_i, \tilde{\mathbf{z}}_i) = \frac{\mathbf{z}_i(\tilde{\mathbf{z}}_i)^T}{\|\mathbf{z}_i\|\|\tilde{\mathbf{z}}_i\|} \,, \tag{5}$$

where $\mathbf{z}$ and $\tilde{\mathbf{z}}_i$ are the i-th node embeddings in Z and $\tilde{Z}$ respectively. A higher similarity denotes that the target node can be well predicted by its neighbors and indicates the target is normal with a higher probability. Therefore, we directly utilize the similarity to measure the normal score of the nodes.

GNN Pre-training. To build generalist models, we pre-train the above neighborhood aggregation network via graph contrastive learning due to the ability to obtain robust and transferable models [You *et al.*, 2020; Zhu *et al.*, 2020] across graphs. Without pre-training, the dataset-specific knowledge would be captured by the model as it is directly optimized based on the neighborhood-based latent attribute prediction of normal and abnormal nodes, limiting the generalizability of the model to other graphs.

Neighborhood Prompting via Latent Attribute Prediction. After the pre-training, we aim to further learn more generalized normal and abnormal patterns via prompt tuning. Thus, we devise learnable prompts appended to the attributes of the neighboring nodes of the target nodes, namely *neighborhood prompts*, for learning robust and discriminative patterns that can detect anomalous nodes in different unseen graphs without any re-training during inference.

Specifically, neighborhood prompting aims to learn some prompt tokens that help maximize the neighborhood-based latent prediction of normal nodes while minimizing that of abnormal nodes simultaneously. To this end, the prompt is designed as a set of shared and learnable tokens that can be incorporated into the normalized node attributes. Formally, the neighborhood prompts are represented as $P = [\mathbf{p}_1, \ldots, \mathbf{p}_k]^T \in \mathbb{R}^{K \times d'}$ where K is the number of vector-based tokens $\mathbf{p}_i$. For each node in $\mathcal{G} = (A, \bar{X})$, the node attributes in the unified feature space are augmented by the weighted combination of these tokens, with the weights obtained from K learnable linear projections:

$$\hat{\mathbf{x}}_i = \bar{\mathbf{x}}_i + \sum_j^K \alpha_j \mathbf{p}_j \,, \quad \alpha_j = \frac{e^{(\mathbf{w}_j)^T \mathbf{x}_i^t}}{\sum_l^K e^{(\mathbf{w}_l)^T \mathbf{x}_i^t}} \,, \tag{6}$$

where α_j denotes the importance score of the token $\mathbf{p}_j$ in the prompt and $\mathbf{w}_j$ is a learnable projection. For convenience, we denote the graph modified by the graph prompt

as $\tilde{\mathcal{G}} = (A, \bar{X} + P)$. Then, $\tilde{\mathcal{G}}$ is fed into the frozen pre-trained model g to obtain the corresponding aggregated embeddings $\tilde{Z}$ and node latent attributes Z via Eq.(3) and Eq.(4) respectively to measure the attribute predictability. To further enhance the representation discrimination, a transformation layer h is applied on the learned $\tilde{Z}$ and Z to transform them into a more anomaly-discriminative feature space,

$$\tilde{Z} = h(\tilde{Z}) \,, \quad Z = h(Z) \,. \tag{7}$$

The transformed representations are then used to measure the latent node attribute predictability with Eq.(5). To optimize P and h, we employ the following training objective,

$$\min_{P,h} \sum \ell(\mathbf{z}_i, \tilde{\mathbf{z}}_i) \,, \tag{8}$$

where $\ell(\mathbf{z}_i, \tilde{\mathbf{z}}_i) = -\text{sim}(\mathbf{z}_i, \tilde{\mathbf{z}}_i)$ if $\mathbf{y}_i = 0$, and $\ell(\mathbf{z}_i, \tilde{\mathbf{z}}_i) = \text{sim}(\mathbf{z}_i, \tilde{\mathbf{z}}_i)$ if $\mathbf{y}_i = 1$.

3.5 Training and Inference of UNPrompt
Training. Given $\mathcal{G}_{\text{train}}$, a neighborhood aggregation network g is optimized via graph contrastive learning. Then, the neighborhood prompts P and the transformation layer h are optimized to capture the graph-agnostic normal and abnormal patterns while keeping the pre-trained model g frozen. In this way, the transferable knowledge of the pre-trained g is maintained, while the neighborhood prompt learning helps learn the generalized normal and abnormal patterns.

Inference. During inference, given $\mathcal{G}_{\text{test}}^{(i)} \in \mathcal{T}_{\text{test}}$, the node attributes are first aligned. Then, the test graph $\mathcal{G}_{\text{test}}^{(i)}$ is augmented with the learned neighborhood prompt P and fed into the model g and the transformation layer h to obtain the neighborhood aggregated representations and the latent node attributes. Finally, the similarity (Eq.(5)) is used as the normal score for anomaly detection. Note that the inference does not require any further re-training and labeled nodes of $\mathcal{G}_{\text{test}}^{(i)}$.

4 Time Complexity Analysis
UNPrompt consists of GNN pre-training, neighborhood prompt learning, and the transformation layer. The overall time complexity is $\mathcal{O}(4E_1(|A|d_h + Nd_hd' + 6Nd_h^2) + 2E_2(|A|d_h + Nd_hd' + 2KNd' + Nd_h^2))$, where d_h is the number of hidden units, $|A|$ returns of the number of edges of the $\mathcal{G}_{\text{train}}$, N is the number of nodes, d' represents the predefined dimensionality of node attributes, E_1 is the number of pre-training epoch, K is the size of prompt and E_2 is the training epochs for prompt learning. Theoretical and empirical results on computational cost are presented in the supplementary[1].

5 Experiments
5.1 Performance on Zero-shot Generalist GAD
Datasets. We evaluate UNPrompt on several real-world GAD datasets from diverse social networks, online shopping co-review networks, and co-purchase networks. Specifically, the social networks include Facebook [Xu *et al.*,

---

[1]https://github.com/mala-lab/UNPrompt/blob/main/UNPrompt_Supplementary.pdf

Metric	Method	Dataset							Avg.
		Amazon	Reddit	Weibo	YelpChi	Aamzon-all	YelpChi-all	Disney	
AUROC	*Unsupervised Methods*								
	AnomalyDAE (ICASSP'20)	$0.5818_{\pm0.039}$	$0.5016_{\pm0.032}$	$\underline{0.7785}_{\pm0.058}$	$0.4837_{\pm0.094}$	$0.7228_{\pm0.023}$	$0.5002_{\pm0.018}$	$0.4853_{\pm0.003}$	$\underline{0.5791}$
	CoLA (TNNLS'21)	$0.4580_{\pm0.054}$	$0.4623_{\pm0.005}$	$0.3924_{\pm0.041}$	$0.4907_{\pm0.017}$	$0.4091_{\pm0.052}$	$0.4879_{\pm0.010}$	$0.4696_{\pm0.065}$	0.4529
	HCM-A (ECML-PKDD'22)	$0.4784_{\pm0.005}$	$0.5387_{\pm0.041}$	$0.5782_{\pm0.048}$	$0.5000_{\pm0.000}$	$0.5056_{\pm0.059}$	$0.5023_{\pm0.005}$	$0.2014_{\pm0.015}$	0.4721
	TAM (NeurIPS'23)	$0.4720_{\pm0.005}$	$\mathbf{0.5725}_{\pm0.004}$	$0.4867_{\pm0.028}$	$0.5035_{\pm0.014}$	$0.7543_{\pm0.002}$	$0.4216_{\pm0.002}$	$0.4773_{\pm0.003}$	0.5268
	GADAM (ICLR'24)	$\underline{0.6646}_{\pm0.063}$	$0.4532_{\pm0.024}$	$0.3652_{\pm0.052}$	$0.3376_{\pm0.012}$	$0.5959_{\pm0.080}$	$0.4829_{\pm0.016}$	$0.4288_{\pm0.023}$	0.4755
	Supervised Methods								
	GCN (ICLR'17)	$0.5988_{\pm0.016}$	$\underline{0.5645}_{\pm0.000}$	$0.2232_{\pm0.074}$	$0.5366_{\pm0.019}$	$0.7195_{\pm0.002}$	$\underline{0.5486}_{\pm0.001}$	$0.5000_{\pm0.000}$	0.5273
	GAT (ICLR'18)	$0.4981_{\pm0.008}$	$0.5000_{\pm0.025}$	$0.4521_{\pm0.101}$	$\underline{0.5871}_{\pm0.016}$	$0.5005_{\pm0.012}$	$0.4802_{\pm0.004}$	$0.5175_{\pm0.054}$	0.5051
	BWGNN (ICML'22)	$0.4769_{\pm0.020}$	$0.5208_{\pm0.016}$	$0.4815_{\pm0.108}$	$0.5538_{\pm0.027}$	$0.3648_{\pm0.050}$	$0.5282_{\pm0.015}$	$0.6073_{\pm0.026}$	0.5048
	GHRN (WebConf'23)	$0.4560_{\pm0.033}$	$0.5253_{\pm0.006}$	$0.5318_{\pm0.038}$	$0.5524_{\pm0.020}$	$0.3382_{\pm0.085}$	$0.5125_{\pm0.016}$	$0.5336_{\pm0.030}$	0.4928
	XGBGraph (NeurIPS'23)	$0.4179_{\pm0.000}$	$0.4601_{\pm0.000}$	$0.5373_{\pm0.000}$	$0.5722_{\pm0.000}$	$\underline{0.7950}_{\pm0.000}$	$0.4945_{\pm0.000}$	$\mathbf{0.6692}_{\pm0.000}$	0.5637
	GraphPrompt (WebConf'23)	$0.4904_{\pm0.001}$	$0.4677_{\pm0.001}$	$0.6135_{\pm0.008}$	$0.3935_{\pm0.002}$	$0.3215_{\pm0.001}$	$0.4976_{\pm0.000}$	$0.5192_{\pm0.002}$	0.4719
	UNPrompt (Ours)	$\mathbf{0.7525}_{\pm0.016}$	$0.5337_{\pm0.002}$	$\mathbf{0.8860}_{\pm0.007}$	$\mathbf{0.5875}_{\pm0.016}$	$0.7962_{\pm0.022}$	$\mathbf{0.5558}_{\pm0.012}$	$\underline{0.6412}_{\pm0.030}$	$\mathbf{0.6790}$
AUPRC	*Unsupervised Methods*								
	AnomalyDAE (ICASSP'20)	$0.0833_{\pm0.015}$	$0.0327_{\pm0.004}$	$\underline{0.6064}_{\pm0.031}$	$0.0624_{\pm0.017}$	$0.1921_{\pm0.026}$	$0.1484_{\pm0.009}$	$0.0566_{\pm0.000}$	$\underline{0.1688}$
	CoLA (TNNLS'21)	$0.0669_{\pm0.002}$	$0.0391_{\pm0.004}$	$0.1189_{\pm0.014}$	$0.0511_{\pm0.000}$	$0.0861_{\pm0.019}$	$0.1466_{\pm0.003}$	$0.0701_{\pm0.023}$	0.0827
	HCM-A (ECML-PKDD'22)	$0.0669_{\pm0.002}$	$0.0391_{\pm0.004}$	$0.1189_{\pm0.014}$	$0.0511_{\pm0.000}$	$0.0861_{\pm0.019}$	$0.1466_{\pm0.003}$	$0.0355_{\pm0.001}$	0.0777
	TAM (NeurIPS'23)	$0.0666_{\pm0.001}$	$\underline{0.0413}_{\pm0.001}$	$0.1240_{\pm0.014}$	$0.0524_{\pm0.002}$	$0.1736_{\pm0.004}$	$0.1240_{\pm0.001}$	$0.0628_{\pm0.001}$	0.0921
	GADAM (ICLR'24)	$0.1562_{\pm0.103}$	$0.0293_{\pm0.001}$	$0.0830_{\pm0.005}$	$0.0352_{\pm0.001}$	$0.1595_{\pm0.121}$	$0.1371_{\pm0.006}$	$0.0651_{\pm0.012}$	0.0951
	Supervised Methods								
	GCN (ICLR'17)	$\underline{0.0891}_{\pm0.007}$	$\mathbf{0.0439}_{\pm0.000}$	$0.1109_{\pm0.020}$	$0.0648_{\pm0.009}$	$0.1536_{\pm0.002}$	$\underline{0.1735}_{\pm0.000}$	$0.0484_{\pm0.000}$	0.0977
	GAT (ICLR'18)	$0.0688_{\pm0.002}$	$0.0329_{\pm0.002}$	$0.1009_{\pm0.017}$	$\mathbf{0.0810}_{\pm0.005}$	$0.0696_{\pm0.001}$	$0.1400_{\pm0.002}$	$0.0530_{\pm0.004}$	0.0780
	BWGNN (ICML'22)	$0.0652_{\pm0.002}$	$0.0389_{\pm0.003}$	$0.2241_{\pm0.046}$	$0.0708_{\pm0.016}$	$0.0586_{\pm0.005}$	$0.1605_{\pm0.005}$	$0.0624_{\pm0.003}$	0.0972
	GHRN (WebConf'23)	$0.0633_{\pm0.003}$	$0.0407_{\pm0.002}$	$0.1965_{\pm0.059}$	$0.0661_{\pm0.010}$	$0.0569_{\pm0.006}$	$0.1505_{\pm0.004}$	$0.0519_{\pm0.003}$	0.0894
	XGBGraph (NeurIPS'23)	$0.0536_{\pm0.000}$	$0.0330_{\pm0.000}$	$0.2256_{\pm0.000}$	$0.0655_{\pm0.000}$	$\underline{0.2307}_{\pm0.000}$	$0.1449_{\pm0.000}$	$\underline{0.1215}_{\pm0.000}$	0.1250
	GraphPrompt (WebConf'23)	$0.0661_{\pm0.000}$	$0.0334_{\pm0.000}$	$0.2014_{\pm0.004}$	$0.0382_{\pm0.000}$	$0.0666_{\pm0.000}$	$0.1542_{\pm0.000}$	$0.0617_{\pm0.001}$	0.0888
	UNPrompt (Ours)	$\mathbf{0.1602}_{\pm0.013}$	$0.0351_{\pm0.000}$	$\mathbf{0.6406}_{\pm0.026}$	$\underline{0.0712}_{\pm0.008}$	$\mathbf{0.2430}_{\pm0.028}$	$\mathbf{0.1810}_{\pm0.012}$	$\mathbf{0.1236}_{\pm0.031}$	$\mathbf{0.2078}$

Table 1: AUROC and AUPRC results on seven real-world GAD datasets with the models trained on Facebook only. For each dataset, the best performance per column within each metric is boldfaced, with the second-best underlined. "Avg" denotes the average performance.

2022], Reddit [Kumar *et al.*, 2019] and Weibo [Kumar *et al.*, 2019]. The co-review networks consist of Amazon [McAuley and Leskovec, 2013], YelpChi [Rayana and Akoglu, 2015], Amazon-all and YelpChi-all. Disney [Sánchez *et al.*, 2013] is a co-purchase network.

Competing Methods. Since there is no zero-shot generalist GAD method, a set of state-of-the-art (SotA) unsupervised and supervised competing methods are employed for comparison. The unsupervised methods comprise reconstruction-based AnomalyDAE [Fan *et al.*, 2020], contrastive learning-based CoLA [Liu *et al.*, 2021b], hop prediction-based HCM-A [Huang *et al.*, 2022], local affinity-based TAM [Qiao and Pang, 2023] and GADAM [Chen *et al.*, 2024]. Supervised methods include two conventional GNNs – GCN [Kipf and Welling, 2016] and GAT [Veličković *et al.*, 2017] – and three SotA GAD GNNs – BWGNN [Tang *et al.*, 2022], GHRN [Gao *et al.*, 2023] and XGBGraph [Tang *et al.*, 2023]. As a graph prompting method, GraphPrompt [Liu *et al.*, 2023c] is also used. Following [Liu *et al.*, 2024; Qiao and Pang, 2023; Qiao *et al.*, 2024], two widely-used metrics, AUROC and AUPRC, are used to evaluate the performance of all methods. For both metrics, the higher value denotes the better performance. Moreover, for each method, we report the average performance with standard deviations after 5 independent runs with different random seeds.

Implementation Details. For a fair comparison, the common dimensionality is set to eight for all methods, and SVD is used for feature projection. The number of GNN layers is set to one and the number of hidden units is 128. The transformation layer is implemented as a one-layer MLP with the same number of hidden units. The size of the neighborhood prompt is set to one. Results for other hyperparameter settings are presented in the supplementary. For all

baselines, their recommended optimization and hyperparameter settings are used. Note that GraphPrompt is a graph prompting method. To adapt it for generalist GAD, after pre-training, we further perform the prompt learning in the source graph for supervised GAD. Then, the pre-trained model and learned prompts are used for generalist anomaly detection. UNPrompt and all competing methods are trained on Facebook and then directly tested on the other GAD datasets. Facebook is used in the training since its anomaly patterns are more generic.

Main Results. The results of all methods are presented in Table 1 and we can have the following observations. (1) Under the proposed generalist GAD scenario where a model is trained on a single dataset and evaluated on seven other datasets, all the competing baselines fail to work well, demonstrating that it is very challenging to build a generalist GAD model that generalizes across different datasets under the zero-shot setting. (2) For supervised methods, the simple GCN achieves better performance than the specially designed GAD GNNs. This can be attributed to more dataset-specific knowledge being captured in these specialized GAD models, limiting their generalization capacity to the unseen testing graphs. (3) Unsupervised methods perform more stably than supervised methods across the target graphs. This is because the unsupervised objectives are closer to the shared anomaly patterns across graphs compared to the supervised ones, especially for TAM which employs a fairly generalized local affinity-based objective for GAD. (4) Despite being a graph prompting method, GraphPrompt fails to achieve promising performance for zero-shot generalist GAD. This is because no label information is provided and no further model optimization is performed during the inference. (5) The proposed UN-Prompt demonstrates strong and stable generalist GAD ca-

Method	Amazon	Reddit	Weibo	YelpChi	Avg.
UNPrompt	**0.7525**	0.5337	**0.8860**	**0.5875**	**0.6899**
w/o Normalization	0.4684	0.5006	0.1889	0.5620	0.4300
w/o Pre-training	0.5400	0.5233	0.5658	0.4672	0.5241
w/o Prompt	0.5328	0.5500	0.4000	0.4520	0.4837
w/o Transformation	0.7331	**0.5556**	0.7406	0.5712	0.6501

Table 2: AUROC results of UNPrompt and its four variants.

pacity across graphs from different distributions and domains. Specifically, UNPrompt achieves the best AUROC performance on 5 out of 7 datasets, and the average performance outperforms the best-competing method by over 9%. The superiority is attributed to i) the proposed coordinate-wise normalization effectively aligns the features across graphs, and ii) the shared generalized normal and abnormal patterns are well captured in the neighborhood prompts.

Ablation Study. To evaluate the importance of each component in UNPrompt, we design four variants, *i.e.*, w/o coordinate-wise normalization, w/o graph contrastive learning-based pre-training, w/o neighborhood prompts, and w/o transformation layer. The results of these variants are reported in the Table 2. Due to the page limits, four datasets are used, including Amazon, Reddit, Weibo and YelpChi. From the table, we can see that all four components contribute to the overall superior performance of UNPrompt. More specifically, (1) without the coordinate-wise normalization, the method fails to calibrate the distributions of diverse node attributes into a common space, leading to large performance drop across all datasets. (2) Besides the semantics alignment, the graph contrastive learning-based pre-training ensures our GNN network is transferable to other graphs instead of overfitting to the training graph. As expected, the performance of the variant without pre-training also drops significantly. (3) If the neighborhood prompts are removed, the learning of latent node attribute prediction is ineffective for capturing generalized normal and abnormal patterns. (4) The variant without the transformation layer achieves inferior performance on nearly all the datasets, demonstrating the importance of mapping the features into a more anomaly-discriminative space.

Sensitivity w.r.t the Neighborhood Prompt Size. We evaluate the sensitivity of UNPrompt w.r.t the size of the neighborhood prompts *i.e.*, K. We vary K in the range of $[1, 9]$ and report the results in Figure 3(a). It is clear that the performances on Reddit, Weibo and YelpChi-all remain stable with varying sizes of neighborhood prompts while the other datasets show slight fluctuation, demonstrating that the generalized normal and abnormal patterns can be effectively captured in our neighborhood prompts even with a small size.

Prompt learning using latent attribute prediction vs. alternative graph anomaly measures. To further justify the effectiveness of latent attribute predictability on learning generalized GAD patterns in our prompt learning module, we compare our learnable anomaly measure to the recently proposed anomaly measure, local node affinity [Qiao and Pang, 2023]. All modules of UNPrompt are fixed, with only the latent attribute prediction task replaced as the maximization of local affinity as in TAM. The results are presented in Fig-
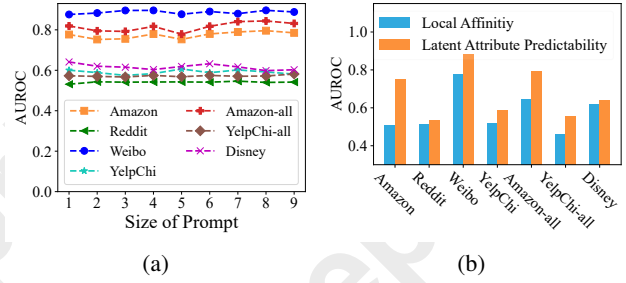

Figure 3: (**a**) AUROC results of UNPrompt w.r.t. varying neighborhood prompt size. (**b**). The AUROC performance of generalist GAD with different prompt learning objectives.

Method	Dataset						
	Amazon	Facebook	Reddit	YelpChi	Amazon-all	YelpChi-all	Avg.
iForest	0.5621	0.5382	0.4363	0.4120	0.1914	0.3617	0.4169
ANOMALOUS	0.4457	0.9021	0.5387	0.4956	0.3230	0.3474	0.5087
DOMINANT	0.5996	0.5677	0.5555	0.4133	0.6937	0.5390	0.5615
CoLA	0.5898	0.8434	<u>0.6028</u>	0.4636	0.2614	0.4801	0.5402
SL-GAD	0.5937	0.7936	0.5677	0.3312	0.2728	0.5551	0.5190
HCM-A	0.3956	0.7387	0.4593	0.4593	0.4191	0.5691	0.5069
ComGA	0.5895	0.6055	0.5453	0.4391	0.7154	0.5352	0.5716
TAM	<u>0.7064</u>	<u>0.9144</u>	0.6023	<u>0.5643</u>	<u>0.8476</u>	<u>0.5818</u>	<u>0.7028</u>
UNPrompt (Ours)	**0.7335**	**0.9379**	**0.6067**	**0.6223**	**0.8516**	**0.6084**	**0.7267**

Table 3: AUROC results of unsupervised GAD methods.

ure 3(b). We can see that the latent attribute predictability consistently and significantly outperforms the local affinity-based measure across all graphs, demonstrating its superiority in learning generalized patterns for generalist GAD.

5.2 Performance in Unsupervised GAD

UNPrompt can also be easily adapted to conventional unsupervised GAD. Experimental details are provided in the supplementary. The results of comparing UNPrompt to SotA unsupervised methods are presented in Table 3. Despite being a generalist method, UNPrompt also works very well as a specialized GAD model. UNPrompt substantially outperforms all the baselines on all datasets. Particularly, the average performance of UNPrompt surpasses the best-competing method by over 2%, showing that the latent node attribute predictability is a generalized GAD measure that also holds for different graphs under unsupervised settings.

6 Conclusion

In this paper, we propose a novel zero-shot generalist GAD method that trains one detector on a single dataset and can effectively generalize to other unseen target graphs without any further re-training or labeled nodes of target graphs during inference. The attribute inconsistency and the absence of generalized anomaly patterns are the main obstacles for generalist GAD. To address these issues, two main modules are proposed, *i.e.*, coordinate-wise normalization-based attribute unification and neighborhood prompt learning. The first module aligns node attribute dimensionality and semantics, while the second module captures generalized normal and abnormal patterns via the neighborhood-based latent node attribute prediction. Extensive experiments on both generalist and unsupervised GAD demonstrate the effectiveness of UNPrompt.

Acknowledgments

Contribution Statement

$\dagger$ denotes that Chaoxi Niu and Hezhe Qiao are co-first authors and contribute equally to this work.

References

[Abdi and Williams, 2010] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[Chai *et al.*, 2022] Ziwei Chai, Siqi You, Yang Yang, Shiliang Pu, Jiarong Xu, Haoyang Cai, and Weihao Jiang. Can abnormality be detected by graph neural networks? In *IJCAI*, pages 1945–1951, 2022.

[Chen *et al.*, 2024] Jingyan Chen, Guanghui Zhu, Chunfeng Yuan, and Yihua Huang. Boosting graph anomaly detection with adaptive message passing. In *The Twelfth International Conference on Learning Representations*, 2024.

[Ding *et al.*, 2019] Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *Proceedings of the 2019 SIAM international conference on data mining*, pages 594–602. SIAM, 2019.

[Ding *et al.*, 2021] Kaize Ding, Kai Shu, Xuan Shan, Jundong Li, and Huan Liu. Cross-domain graph anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2406–2415, 2021.

[Fan *et al.*, 2020] Haoyi Fan, Fengbin Zhang, and Zuoyong Li. Anomalydae: Dual autoencoder for anomaly detection on attributed networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5685–5689. IEEE, 2020.

[Fang *et al.*, 2024] Taoran Fang, Yunchao Zhang, Yang Yang, Chunping Wang, and Lei Chen. Universal prompt tuning for graph neural networks. *Advances in Neural Information Processing Systems*, 36, 2024.

[Gao *et al.*, 2023] Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. Addressing heterophily in graph anomaly detection: A perspective of graph spectrum. In *Proceedings of the ACM Web Conference 2023*, pages 1528–1538, 2023.

[Huang *et al.*, 2022] Tianjin Huang, Yulong Pei, Vlado Menkovski, and Mykola Pechenizkiy. Hop-count based self-supervised anomaly detection on attributed networks. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 225–241. Springer, 2022.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[Kumar *et al.*, 2019] Srijan Kumar, Xikun Zhang, and Jure Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1269–1278, 2019.

[Li *et al.*, 2023] Aodong Li, Chen Qiu, Marius Kloft, Padhraic Smyth, Maja Rudolph, and Stephan Mandt. Zero-shot anomaly detection via batch normalization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[Li *et al.*, 2024] Yuhan Li, Peisong Wang, Zhixun Li, Jeffrey Xu Yu, and Jia Li. Zerog: Investigating cross-dataset zero-shot transferability in graphs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1725–1735, 2024.

[Liu *et al.*, 2021a] Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Pick and choose: a gnn-based imbalanced learning approach for fraud detection. In *Proceedings of the web conference 2021*, pages 3168–3177, 2021.

[Liu *et al.*, 2021b] Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE transactions on neural networks and learning systems*, 33(6):2378–2392, 2021.

[Liu *et al.*, 2022] Kay Liu, Yingtong Dou, Yue Zhao, Xueying Ding, Xiyang Hu, Ruitong Zhang, Kaize Ding, Canyu Chen, Hao Peng, Kai Shu, et al. Bond: Benchmarking unsupervised outlier node detection on static attributed graphs. *Advances in Neural Information Processing Systems*, 35:27021–27035, 2022.

[Liu *et al.*, 2023a] Jiawei Liu, Cheng Yang, Zhiyuan Lu, Junze Chen, Yibo Li, Mengmei Zhang, Ting Bai, Yuan Fang, Lichao Sun, Philip S Yu, et al. Towards graph foundation models: A survey and beyond. *arXiv preprint arXiv:2310.11829*, 2023.

[Liu *et al.*, 2023b] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[Liu *et al.*, 2023c] Zemin Liu, Xingtong Yu, Yuan Fang, and Xinming Zhang. Graphprompt: Unifying pre-training and downstream tasks for graph neural networks. In *Proceedings of the ACM Web Conference 2023*, pages 417–428, 2023.

[Liu *et al.*, 2024] Yixin Liu, Shiyuan Li, Yu Zheng, Qingfeng Chen, Chengqi Zhang, and Shirui Pan. Arc: A generalist graph anomaly detector with in-context learning. *arXiv preprint arXiv:2405.16771*, 2024.

[McAuley and Leskovec, 2013] Julian John McAuley and Jure Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews.

In *Proceedings of the 22nd international conference on World Wide Web*, pages 897–908, 2013.

[Pang *et al.*, 2021] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021.

[Peng *et al.*, 2018] Zhen Peng, Minnan Luo, Jundong Li, Huan Liu, Qinghua Zheng, et al. Anomalous: A joint modeling approach for anomaly detection on attributed networks. In *IJCAI*, volume 18, pages 3513–3519, 2018.

[Qiao and Pang, 2023] Hezhe Qiao and Guansong Pang. Truncated affinity maximization: One-class homophily modeling for graph anomaly detection. *Advances in Neural Information Processing Systems*, 36, 2023.

[Qiao *et al.*, 2024] Hezhe Qiao, Hanghang Tong, Bo An, Irwin King, Charu Aggarwal, and Guansong Pang. Deep graph anomaly detection: A survey and new perspectives. *arXiv preprint arXiv:2409.09957*, 2024.

[Radford *et al.*, 2021] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[Rayana and Akoglu, 2015] Shebuti Rayana and Leman Akoglu. Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining*, pages 985–994, 2015.

[Sánchez *et al.*, 2013] Patricia Iglesias Sánchez, Emmanuel Müller, Fabian Laforet, Fabian Keller, and Klemens Böhm. Statistical selection of congruent subspaces for mining attributed graphs. In *2013 IEEE 13th international conference on data mining*, pages 647–656. IEEE, 2013.

[Stewart, 1993] Gilbert W Stewart. On the early history of the singular value decomposition. *SIAM review*, 35(4):551–566, 1993.

[Sun *et al.*, 2023] Xiangguo Sun, Hong Cheng, Jia Li, Bo Liu, and Jihong Guan. All in one: Multi-task prompting for graph neural networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2120–2131, 2023.

[Tang *et al.*, 2022] Jianheng Tang, Jiajin Li, Ziqi Gao, and Jia Li. Rethinking graph neural networks for anomaly detection. In *International Conference on Machine Learning*, pages 21076–21089. PMLR, 2022.

[Tang *et al.*, 2023] Jianheng Tang, Fengrui Hua, Ziqi Gao, Peilin Zhao, and Jia Li. Gadbench: Revisiting and benchmarking supervised graph anomaly detection. *Advances in Neural Information Processing Systems*, 36:29628–29653, 2023.

[Veličković *et al.*, 2017] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[Wang *et al.*, 2023] Qizhou Wang, Guansong Pang, Mahsa Salehi, Wray Buntine, and Christopher Leckie. Cross-domain graph anomaly detection via anomaly-aware contrastive alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4676–4684, 2023.

[Wu *et al.*, 2019] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. PMLR, 2019.

[Wu *et al.*, 2020] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2020.

[Xu *et al.*, 2022] Zhiming Xu, Xiao Huang, Yue Zhao, Yushun Dong, and Jundong Li. Contrastive attributed network anomaly detection with data augmentation. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 444–457. Springer, 2022.

[Yang *et al.*, 2019] Yang Yang, Yuhong Xu, Yizhou Sun, Yuxiao Dong, Fei Wu, and Yueting Zhuang. Mining fraudsters and fraudulent strategies in large-scale mobile social networks. *IEEE Transactions on Knowledge and Data Engineering*, 33(1):169–179, 2019.

[You *et al.*, 2020] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

[Zhao *et al.*, 2024] Haihong Zhao, Aochuan Chen, Xiangguo Sun, Hong Cheng, and Jia Li. All in one and one for all: A simple yet effective method towards cross-domain graph pretraining. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4443–4454, 2024.

[Zheng *et al.*, 2021] Yu Zheng, Ming Jin, Yixin Liu, Lianhua Chi, Khoa T Phan, and Yi-Ping Phoebe Chen. Generative and contrastive self-supervised learning for graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12220–12233, 2021.

[Zhou *et al.*, 2024] Qihang Zhou, Guansong Pang, Yu Tian, Shibo He, and Jiming Chen. AnomalyCLIP: Object-agnostic prompt learning for zero-shot anomaly detection. In *The Twelfth International Conference on Learning Representations*, 2024.

[Zhu and Pang, 2024] Jiawen Zhu and Guansong Pang. Toward generalist anomaly detection via in-context residual learning with few-shot sample prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17826–17836, 2024.

[Zhu *et al.*, 2020] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep graph contrastive representation learning. *arXiv preprint arXiv:2006.04131*, 2020.